Syncro: A Universal Reverse-Auction Marketplace for Goods and Services

Problem Domain

In traditional e-commerce, buyers spend significant time searching for specific goods (like 50 custom cakes) or specialized services (like a Physics tutor) and comparing prices manually. Sellers, on the other hand, often struggle to find active leads who are ready to buy immediately. There is a "discovery gap" where the buyer has a specific need, but no easy way to have multiple sellers compete for their business in real-time.

Proposed Solution

Syncro is a "Reverse Auction" platform that flips the traditional marketplace model. Instead of sellers listing items, **Clients** post a "Request for Proposal" . Verified **Sellers** (caterers, tutors, freelancers) receive notifications based on their specialties and submit competitive bids. This ensures the client gets the best market price while sellers gain access to high-intent customers.

Core Functionalities

- **Dual-Role Authentication:** Users can register and switch between 'Client' and 'Seller' profiles.
- **Smart Request Posting:** Clients can post detailed requirements, including category (e.g., Education, Food & Beverage), quantity, date, and budget range.
- **Niche-Based Notifications:** The system automatically alerts sellers only when a request matches their predefined specialty (e.g., a Chemistry tutor won't see cake requests).
- **Competitive Bidding System:** Sellers can view current requests and submit a "Bid" (Price + Message).
- **Bid Management Dashboard:** Clients can view all received bids in a comparison table, view seller ratings, and "Accept" the best offer.
- **Status Tracking:** Real-time updates as a project moves from *Open* to *Bidding*, *Accepted*, and *Completed*.

Technologies Planned

- **Frontend: React.js** or **Next.js** for a dynamic, responsive user interface.
- **Backend: Node.js (Express)** or **Python (FastAPI)** to handle business logic and notifications.
- **Database: PostgreSQL** or **MySQL** (Relational DBMS) to manage complex links between Users, Specialties, Requests, and Bids.
- **Real-time Logic: WebSockets (Socket.io)** for instant bid notifications.

Expected Deliverables

1. **Web Application:** A fully functional frontend and backend system.
2. **Database Schema:** A normalized E-R diagram and SQL initialization scripts.
3. **API Documentation:** Documentation of endpoints for posting requests and submitting bids.
4. **User Manual:** A guide for clients and sellers on how to use the auction system.